Problem 1217 : Minimum Cost to Move Chips to the Same Position

<https://leetcode.com/problems/minimum-cost-to-move-chips-to-the-same-position/>

My Solution :

<https://leetcode.com/problems/minimum-cost-to-move-chips-to-the-same-position/discuss/925454/Simple-Python-3-Solution-Runtime-beats-86.88>

1. Iterate through the list of position and count the number of odd elements and even elements.
2. If the number of even elements is greater, then move the odd elements by 0 or even number of steps + 1 step. So cost = 1 \* number of odd elements.
3. If the number of odd elements is greater, then move all the even elements by 1 step or 1 step + even number of steps. The even number of steps have no cost. So cost = 1 \* number of even elements.

class Solution:

def minCostToMoveChips(self, position: List[int]) -> int:

count\_odd = 0

count\_even = 0

for i, pos in enumerate(position):

if pos % 2 == 0:

count\_even += 1

else:

count\_odd += 1

if count\_odd < count\_even:

return(count\_odd)

else:

return(count\_even)